CTE203 Data Structures
Syllabus

Course Details
Course Name: CTE203 – Data Structures
Course Credits: 4
ECTS Credits: 6
Prerequisite: CTE102

Instructor: İpek Sözen
E-mail: sozen@ctp.bilkent.edu.tr
Room, Phone: GZ-18, 5067
Office Hours: Monday 09:40 – Wednesday 10:40

Assistant: Ergem Özdemir
E-mail: eozdemir@ctp.bilkent.edu.tr
Room, Phone: GZ-58, 3351
Office Hours: Tuesday 09:40, 10:40

Lecture hours and place: Wednesday 8:40 - 9:40 GZ-55
Friday 10:40 - 11:40 GZ-55
Lab hours and place: Wednesday 13:40 - 14:40
URL (web pages): http://www.cte.bilkent.edu.tr/~cte203/

Course Description

Aim
The aim of the course is to teach students abstract data types and dynamic data structures so that they can be able to develop moderately complex C programs that uses the introduced data structures.

Learning Outcomes
Upon completion of the course, students will be able to:
- Represent and model real world aggregations of data using structures, linked lists, stacks, and queues in the C language.
- Develop efficient moderately complex computer programs by using appropriate structures.
- Determine the most appropriate abstract data type by classifying and comparing them.
- Create recursive algorithms for implementing applicable programs.
- Design and implement small C programs that effectively use I/O concepts (from/to text and binary files).

Course Outline:
- Strings
- Files
- Structures
- Linked Lists
- Recursion
- Stacks and Queues
Text Book:
• Problem Solving and Program Design in C, Hanly, Koffman, Addison Wesley

Reference Books:
• Lecture notes prepared by the instructor;
• Web references/resources.

Assessments & Grading:
• Quizzes: (10%)
• Lab Performance: (10%)
• Lab Exam: (30%)
• Midterm: (20%)
• Final: (30%)

Attendance:
Attendance is an important criteria in following the course topics and being successful at the end of the semester.

Weekly outline:

<table>
<thead>
<tr>
<th>Week</th>
<th>Topics to be covered</th>
<th>Lab. Objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Jan 29</td>
<td>Information about the course.</td>
</tr>
<tr>
<td>3</td>
<td>Feb 08 - 12</td>
<td>Introduction to files. Text files: declaring, opening, verifying existence, checking for end-of-file, reading from and writing into text files.</td>
</tr>
<tr>
<td>4</td>
<td>Feb 15 - 19</td>
<td>Binary files: creating, reading from, writing into.</td>
</tr>
<tr>
<td>5</td>
<td>Feb 22 - 26</td>
<td>QUIZ 1. Structures: Introduction, declaring, initializing, i/o. Structure type as function parameters.</td>
</tr>
<tr>
<td>6</td>
<td>Mar 01 - 05</td>
<td>Structures: Arrays of structures. Pointers to structures. Structures as output parameters. Nested structures. Structures with array elements.</td>
</tr>
<tr>
<td>7</td>
<td>Mar 08 - 12</td>
<td>Structures: examples. Writing and reading structures to/from binary files.</td>
</tr>
<tr>
<td>8</td>
<td>Mar 15 - 19</td>
<td>MIDTERM WEEK</td>
</tr>
<tr>
<td>9</td>
<td>Dynamic memory allocation. Cast operator.</td>
<td>Creating a linked list</td>
</tr>
<tr>
<td>Date</td>
<td>Topics</td>
<td>Notes</td>
</tr>
<tr>
<td>--------------</td>
<td>------------------------------------------------------------------------</td>
<td>----------------------------</td>
</tr>
<tr>
<td>Mar 22 - 26</td>
<td>Linked lists: creating linked lists, adding nodes to the beginning and end, searching for values.</td>
<td>• Adding a node to the beginning and end of the list, searching a node in a linked list</td>
</tr>
<tr>
<td>10 Mar 29–Apr 02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11 Apr 05 – 09</td>
<td><strong>SPRING RECESS</strong></td>
<td>No Lab</td>
</tr>
<tr>
<td>12 Apr 12 – 16</td>
<td>Linked lists: Inserting and deleting nodes. Examples with linked lists.</td>
<td>• Inserting, deleting nodes to a linked list</td>
</tr>
<tr>
<td>13 Apr 19 - 22</td>
<td>Linked lists: more examples</td>
<td>Linked List Implementation</td>
</tr>
<tr>
<td>14 Apr 26 - 30</td>
<td>Recursion. Recursive functions.</td>
<td>• Implementation of recursive functions.</td>
</tr>
<tr>
<td>15 May 03 - 07</td>
<td><strong>QUIZ 2.</strong> Stacks.</td>
<td>• Stack implementation</td>
</tr>
<tr>
<td>16 May 10 - 14</td>
<td>Queues.</td>
<td>• Queue implementation</td>
</tr>
</tbody>
</table>